

WHAT IS CLAIMED IS:

1 1. An XML (extensible markup language) transaction definition document stored
2 on a computer-readable medium comprising a plurality of operation data portions each
3 defining an operation; the plurality of operations collectively defining a transaction;
4 each operation data portion, when parsed by a process automation application, causing
5 the process automation application to communicate with a service application program
6 to perform the operation; at least one operation data portion comprising a conditional
7 logic data portion; when parsed by the process automation application, the conditional
8 logic data portion causing the process automation application to condition
9 performance of a next operation on evaluation of operation response data from
10 performing the operation.

1 2. The XML transaction definition document of claim 1 wherein the conditioning
2 logic data portion indicates at least one of a mathematical expression, a function, and a
3 variable data item; and wherein, when parsed by the process automation application,
4 the conditioning logic data portion causes the process automation application to
5 evaluate the at least one of the mathematical expression, the function, and the variable
6 data item using the operation response data.

1 3. The XML transaction definition document of claim 1 wherein the XML
2 transaction definition document is represented as a directed acyclic graph data
3 structure; and wherein the plurality of operation data portions are represented as nodes
4 in the directed acyclic graph.

1 4. The XML transaction definition document of claim 1 wherein the at least one
2 operation data portion further comprises an input data portion including an input
3 document identifier indicating a source XML document; the source XML document
4 including input data for performing the operation.

1 5. The XML transaction definition document of claim 4 wherein the input
2 document identifier indicates an XML output response document as the source XML
3 document; the XML output response document being produced by a service
4 application previously performing one of the plurality of operations collectively
5 defining the transaction.

1 6. The XML transaction definition document of claim 1 wherein the at least one
2 operation data portion further comprises an input data portion indicating argument
3 input data for performing the operation; the argument input data including a variable
4 name, an argument value, and an argument expression; the argument expression, when
5 parsed and evaluated by the process automation application, producing the argument
6 value.

1 7. The XML transaction definition document of claim 6 wherein the argument
2 expression indicates at least one of a mathematical expression, a function, and a
3 variable data item; and wherein, when parsed by the process automation application,
4 the input data portion causes the process automation application to evaluate the at least
5 one of the mathematical expression, the function, and the variable data item to produce
6 the argument value.

1 8. An XML (extensible markup language) transaction definition document stored
2 on a computer-readable medium comprising a plurality of operation data portions each
3 defining an operation; the plurality of operations collectively defining a transaction;
4 each operation data portion, when parsed by a process automation application, causing
5 the process automation application to communicate with a service application program
6 to perform the operation; at least one operation data portion indicating a broadcast
7 operation and including a broadcast data portion; when parsed by the process
8 automation application, the broadcast data portion causing the process automation
9 application to communicate with a plurality of service applications to cause each
10 service application to perform the operation.

1 9. The XML transaction definition document of claim 8 wherein the broadcast data
2 portion includes a response data portion; when parsed by the process automation
3 application, the response data portion causing the process automation application to
4 evaluate operation response data received from the plurality of service applications to
5 determine a success outcome or a failure outcome of the broadcast operation.

1 10. The XML transaction definition document of claim 9 wherein the response data
2 portion indicates a minimum response value indicating a minimum number of required
3 responses from service applications performing the operation; when parsed by the
4 process automation application, the response data portion causing the process
5 automation application to determine whether the minimum number of required
6 responses were received from the plurality of service applications.

1 11. The XML transaction definition document of claim 9 wherein the broadcast data
2 portion further includes an expression data portion indicating at least one of a
3 mathematical expression, a function, and a variable data item; when parsed by the
4 process automation application, the expression data portion causing the process
5 automation application to evaluate the at least one of the mathematical expression, the
6 function, and the variable data item using the operation response data to determine the
7 success or failure outcome of the broadcast operation.

1 12. The XML transaction definition document of claim 8 wherein the XML
2 transaction definition document is represented as a directed acyclic graph data
3 structure; and wherein the plurality of operation data portions are represented as nodes
4 in the directed acyclic graph.

1 13. A transaction definition data structure stored on a computer-readable medium
2 comprising a plurality of operation data portions indicating a plurality of operations
3 collectively defining a transaction; each operation data portion defining an operation
4 and comprising:

5 an operation identifier uniquely identifying the operation among the plurality of
6 operations;

7 a service application name indicating a service application for performing the
8 operation;

9 an input data portion indicating input data used by the service operation for
10 performing the operation; and

11 a conditional logic data portion indicating evaluation data conditioning performance of
12 the next operation on evaluation of operation response data received from the
13 service application performing the operation.

1 14. The transaction definition data structure of claim 13 wherein the conditioning
2 logic data portion indicates at least one of a mathematical expression, a function, and a
3 variable data item for use in evaluating the operation response data.

1 15. The transaction definition data structure of claim 13 wherein the input data
2 indicates output response data produced by a service application previously
3 performing one of the plurality of operations collectively defining the transaction.

1 16. The transaction definition data structure of claim 13 wherein the input data
2 portion includes argument input data for performing the operation; the argument input
3 data including a variable name known to the service operation performing the
4 operation, an argument value to be assigned to the variable name, and an argument
5 expression data portion; the argument expression data portion indicating at least one of
6 a mathematical expression, a function, and a variable data item; the at least one of the
7 mathematical expression, the function, and the variable data item being evaluated to
8 produce the argument value as input to the service application performing the
9 operation.

1 17. The transaction definition data structure of claim 13 wherein each operation
2 data portion further comprises an operation link data portion including at least one
3 operation identifier indicating a next operation to be performed subsequent to
4 performing the operation; all of the operation link data portions included in the
5 transaction definition data structure collectively indicating a sequence of operations
6 defining the transaction.

1 18. The transaction definition data structure of claim 17 wherein the operation link
2 data portion includes a plurality of operation identifiers indicating a plurality of next
3 operations to be concurrently performed subsequent to performing the operation.

1 19. The transaction definition data structure of claim 17 wherein a next operation
2 indicated by the sequence of operations collectively indicated by all of the operation
3 link data portions may be changed by a conditional logic data portion of a prior
4 operation evaluating operation response data received from the service application
5 performing the prior operation.

1 20. The transaction definition data structure of claim 13 wherein the transaction
2 definition data structure is an XML document.

1 21. A computer-implemented method for performing a transaction comprising the
2 steps of:

3 producing a transaction instance data structure indicating a plurality of operations
4 constituting a transaction; the transaction instance data structure indicating a
5 linking of the plurality of operations to indicate an operation performance order;
6 the transaction instance data structure further indicating conditioning logic data
7 for changing the operation performance order such that the plurality of
8 operations is capable of being performed in more than one possible order; and
9 for each of the plurality of operations,

Sub A¹⁰ 7

11 producing an operation request message indicating input data for performing an
12 operation;
13 sending the operation request message to a service application to perform the
14 operation using the input data;
15 receiving an operation response message from the service application indicating
16 output data from the operation; and
17 determining a next operation to perform using the conditioning logic data and the
output data of the operation response message.

1 22. The computer-implemented method of claim 21 for performing a transaction
2 wherein the conditioning logic data indicates at least one of a mathematical
3 expression, a function, and a variable data item; and wherein the step of determining
4 the next operation to perform using the conditioning logic data and the output data of
5 the operation response message includes using the output data to evaluate the at least
6 one of the mathematical expression, the function, and the variable data item.

1 23. The computer-implemented method of claim 21 for performing a transaction
2 wherein the operation request message and the operation response message include
3 extensible markup language (XML) tags indicating data items.

Sub A67
1 24. The computer-implemented method of claim 21 for performing a transaction
2 wherein the transaction instance data structure is a directed acyclic graph (DAG)
3 including a plurality of nodes; each operation being represented by a node; the nodes
4 being arranged in the transaction instance DAG such that paths through the transaction
5 instance DAG indicate the more than one possible order in which the plurality of
6 operations may be performed; and wherein performing the transaction further includes
7 traversing a path through the plurality of nodes of the transaction instance DAG.

1 25. The computer-implemented method of claim 24 for performing a transaction
2 wherein the path through the graph is determined at runtime.

1 26. The computer-implemented method of claim 21 for performing a transaction
2 further including receiving a transaction request message indicating a request to
3 perform the transaction from a requesting application residing on a first computer
4 included in a distributed network; and wherein the service application resides on a
5 second computer included in the distributed network.

1 27. The computer-implemented method of claim 26 wherein the distributed
2 network is the Internet.

Sub A67

1 28. An article of manufacture comprising a data storage medium having computer
2 readable instruction data embodied therein; the computer readable instruction data
3 indicating instructions executed by a processor in a processor-controlled machine for
4 managing transaction processing message flow among a plurality of requesting
5 application programs and service application programs resident on a plurality of
6 processor-controlled machines in a distributed network; the computer readable
7 instructions in the article of manufacture comprising:
8 a first portion of instructions which when executed causes the processor to produce a
9 transaction instance data structure indicating a plurality of operations
10 constituting a transaction; the transaction instance data structure indicating a
11 linking of the plurality of operations to indicate an order of execution; the
12 transaction instance data structure further indicating conditioning logic data
13 conditioning execution of at least one operation such that the plurality of
14 operations is capable of being performed in more than one possible order; and
15 a second portion of instructions which when executed causes the processor, for each of
16 the plurality of operations, to produce an operation request message indicating
17 input data for performing an operation, to send the operation request message to
18 a service application to perform the operation using the input data, to receive an
19 operation response message from the service application indicating output data
20 from the operation, and to determine a next operation to perform using the
21 conditioning logic data and the output data of the operation response message.

Sub AS 7

1 29. The article of claim 28 wherein the conditioning logic data indicates at least
2 one of a mathematical expression, a function, and a variable data item; and wherein
3 the second portion of instructions further includes a third portion of instructions
4 which, when executed, causes the processor, for each of the plurality of operations, to
5 use the output data to evaluate the at least one of the mathematical expression, the
6 function, and the variable data item in order to determine the next operation to
7 perform.

1 30. The article of claim 28 wherein the transaction instance data structure is a
2 directed acyclic graph (DAG) including a plurality of nodes; each operation being
3 represented by a node; the nodes being arranged in the transaction instance DAG such
4 that paths through the transaction instance DAG indicate the more than one possible
5 order in which the plurality of operations may be performed; and wherein the article
6 further includes a third portion of instructions which, when executed, causes the
7 processor to traverse a path through the plurality of nodes of the transaction instance
8 DAG.

1 31. A computer-implemented method for performing a transaction in a distributed
2 computer network comprising the steps of:
3 receiving a transaction request message from a requesting application program
4 indicating transaction data; the requesting application program residing on a first
5 computer in a distributed computer network;

6 obtaining a transaction definition using the transaction data included in the transaction
7 request message; the transaction definition indicating a plurality of operations
8 and a linking of the plurality of operations to indicate an order for performing the
9 operations; the transaction definition further indicating conditioning logic data
10 conditioning performance of at least one operation such that the plurality of
11 operations is capable of being performed in more than one possible order;
12 producing a transaction instance directed acyclic graph (DAG) using the transaction
13 definition; the transaction instance DAG including a plurality of nodes each
14 indicating one of the plurality of operations and arranged in the transaction
15 instance DAG such that a path through the transaction instance DAG indicates a
16 possible execution order of the plurality of operations;
17 traversing a path through the plurality of nodes of the transaction instance DAG; the
18 path including the operations to be performed; traversing a path through the
19 transaction instance DAG including the steps of
20 producing an operation request message indicating input data for performing an
21 operation;
22 sending the operation request message to a service application to perform the
23 operation using the input data; the service application residing on a second
24 computer in the distributed computer network;
25 receiving an operation response message from the service application indicating
26 output data from the operation; and
27 determining a next operation to perform using the conditioning logic data; and

28 producing a transaction response message using the operation response messages and
29 sending the transaction response message to the requesting application.

1 32. The computer-implemented method of claim 31 for performing a transaction in a
2 distributed computer network wherein
3 the transaction request message indicates a transaction definition name identifying a
4 transaction definition; and
5 the transaction definition is one of a plurality of transaction definitions included in a
6 transaction definition data store; each of the plurality of transaction definitions
7 having a transaction definition name uniquely identifying the transaction
8 definition.

1 33. The computer-implemented method of claim 31 for performing a transaction in a
2 distributed computer network wherein the transaction request message, the operation
3 request message, the operation response message and the transaction response message
4 are XML documents.

1 34. The computer-implemented method of claim 31 for performing a transaction in a
2 distributed computer network wherein no more than one transaction request message is
3 received from the requesting application to initiate a transaction and only one
4 transaction response message is sent to the requesting application indicating output
5 data from performing the transaction.

1 35. A distributed transaction processing system comprising:
2 a plurality of service application programs each capable of performing an operation;
3 a data store including a plurality of transaction definitions; each transaction definition
4 indicating a transaction definition name uniquely identifying the transaction
5 definition and a plurality of operation definitions indicating a plurality of
6 operations constituting a transaction;
7 a requesting application program; the requesting application program producing a
8 transaction request message indicating a transaction definition name identifying
9 one of the plurality of transaction definitions included in the data store; and
10 a computer having a memory device for storing a process automation application; the
11 process automation application receiving the transaction request message
12 indicating the transaction definition name from the requesting application
13 program and using the transaction definition name to obtain the transaction
14 definition from the data store; the process automation application producing an
15 operation request message for each operation definition included in the plurality
16 of operation definitions and sending the operation request messages to at least
17 one service application program;
18 the at least one service application program sending an operation response message
19 indicating an output of performing an operation to the process automation
20 application in response to receiving an operation request message;

21 the process automation application producing a transaction response message using the
22 operation response messages and sending the transaction response message to the
23 requesting application.

1 36. The distributed transaction processing system of claim 35 wherein the
2 transaction definition, the transaction request message, the operation request message,
3 the operation response message and the transaction response message are XML
4 documents.

1 37. The distributed transaction processing system of claim 35 further including a
2 transaction instance directed acyclic graph data structure including a plurality of
3 nodes; the process automation application using the transaction definition to produce
4 the directed acyclic graph data structure; each of the plurality of operation definitions
5 being represented by a node in the directed acyclic graph data structure; and
6 wherein the process automation application traverses a path through the transaction
7 instance directed acyclic graph data structure to process the transaction.

1 38. The distributed transaction processing system of claim 37 wherein the plurality
2 of operation definitions indicate more than one processing order for processing the
3 operations; the transaction instance directed acyclic graph data structure indicating
4 more than one path through the nodes; and wherein the process automation application
5 determines the path to traverse through the directed acyclic graph data structure to
6 process the transaction at runtime.

1 39. The distributed transaction processing system of claim 35 further including a
2 second computer having a memory device for storing at least one of the service
3 applications; the second computer and the computer storing the process automation
4 application being included in a distributed computer network.

1 40. A processor-controlled machine for managing transaction message flow in a
2 distributed computer network; the machine comprising:

3 data communications circuitry connected to a network communications device for
4 receiving signals indicating request messages from at least one remote processor-
5 controlled machine and for sending signals indicating response messages to at
6 least one remote processor-controlled machine;

7 a processor connected for receiving the signals from and for sending the signals to the
8 data communications circuitry; and

9 memory for storing data; the data stored in the memory including

10 instruction data indicating instructions the processor can execute; and

11 transaction definition data; the transaction definition data indicating a plurality of

12 operations and a linking of the plurality of operations to indicate an order

13 for performing the operations; the transaction definition data further

14 indicating conditioning logic data conditioning performance of at least one

15 operation such that the plurality of operations is capable of being

16 performed in more than one possible order;

17 the processor being further connected for accessing the data stored in the memory;

18 the processor, in executing the instructions, receiving signals via the data
19 communications circuitry indicating a transaction request message from a
20 requesting application program indicating transaction data;
21 the processor, further in executing the instructions, obtaining from memory the
22 transaction definition data using the transaction data included in the transaction
23 request message;
24 the processor, further in executing the instructions, producing a transaction instance
25 directed acyclic graph (DAG) using the transaction definition data; the
26 transaction instance DAG including a plurality of nodes each indicating one of
27 the plurality of operations and arranged in the transaction instance DAG such
28 that a path through the transaction instance DAG indicates a possible execution
29 order of the plurality of operations;
30 the processor, still further in executing the instructions, traversing a path through the
31 plurality of nodes of the transaction instance DAG; the path including the
32 operations to be performed; the processor, in traversing a path through the
33 transaction instance DAG,
34 producing an operation request message indicating input data for performing an
35 operation;
36 sending via the data communications circuitry signals indicating the operation
37 request message to a service application to perform the operation using the
38 input data;

39 receiving via the data communications circuitry signals indicating an operation
40 response message from the service application indicating output data from
41 the operation; and
42 determining a next operation to perform using the conditioning logic data;
43 the processor, still further in executing the instructions, producing a transaction
44 response message using the operation response messages and sending the
45 transaction response message to the requesting application via the data
46 communications circuitry.